DATASHEET

VMware vFabric GemFire

Elastic In-Memory Data Management

MODERN DATA MANAGEMENT

- Lightning fast, highly available and scalable data access for modern applications—run your applications 4–40 times faster
- Parallel data execution for extremely high throughput
- Engineered for low latency
- Data replication across nodes and clusters for high availability
- Reliable event notifications—as data changes, applications automatically updated
- Continuous querying updates results with predictably low latency
- "Shared nothing" disk persistence ensures disk failure on one node does not result in data loss
- Data sharing between Java, C++ and C# based applications
- WAN scaling while preserving performance, reliability and data consistency

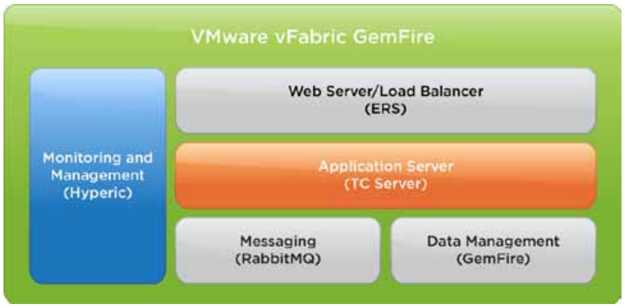
VMware vFabric GemFire
Web Server/Load Balancer (ERS)
Monitoring and Management (Hyperic)
Application Server (TC Server)
Messaging (RabbitMQ)
Data Management (GemFire)

Figure 1: VMware vFabric Cloud Application Platform

Overview

VMware vFabric™ GemFire® provides fast, secure, reliable and scalable access to data in support of modern enterprise and cloud applications. A core component of the VMware vFabric Cloud Application Platform, GemFire enables the delivery of high performance applications that are data rich and continuously available regardless of the number of end-users being served.

As modern applications have become increasingly Web-oriented, data-intensive and inherently more dynamic in nature, the need for modern approaches to data management has evolved. While first-generation Web applications could tolerate time-consuming round-trips to store and retrieve data from a database, newer applications require a different approach, one that is both ideally suited to this new generation of applications but also one that contemplates deployment on top of virtual infrastructure.

GemFire provides the data management capabilities of a database with the performance that only in-memory can provide. It is ideally suited for the needs of modern applications that require real-time access to data and the ability to solve some of the most complex data challenges in the world.

Key Highlights

Bypass high database transaction costs derived from spent CPU cycles, network traffic, database access latency and more by managing data in-memory with GemFire.

Developer Productivity

- Re-engineered APIs simplify development; quickstart programming examples included
- Declaratively configure GemFire data infrastructure from Spring with minimal effort connecting secure, reliable and scalable data to enterprise applications
- Built-in exception handling traverses GemFire to Spring for consistency and can be applied transparently
- Integration with the popular Spring Framework transaction management capabilities makes writing and supporting transactional enterprise applications fast, secure, reliable and scalable
- Hibernate Cache Module provides fast, scalable, distributed L2 caching for applications leverging Hibernate Object-relational Mapping

Database-like Persistence

- "Shared Nothing" parallel disk persistence combines high performance at scale with cluster-wide high availability and sophisticated failure handling
- Synchronous read-through, write-through, or Asynchronous write-behind to backend data sources
- Continuous query support – automatically update applications with data from memory, no roundtrips to a database

Cloud Scale and Operational Efficiency

- Dynamic data partitioning across the system evens out load to enable high scale
- Co-located transactions, 2-3 times higher performance for thin clients
- Wide Area Networking (WAN) support scales across remote sites
- Several management options are available, including a command-line utility, a Java Management Extensions (JMX) Agent, and GFMon, a graphical user interface monitoring tool
- HTTP Session Management Module offloads and manages HTTP session state for tc Server or Tomcat servers. Pre-configured and automatically launching with tc Server, GemFire HTTP Session Management provides high performance and scalability for web applications with high user loads or very large sessions.

Deployment Options

- GemFire runs on a Java Runtime Environment (JRE) in 32-bit and 64-bit mode on Windows, Linux and Solaris.
- Client nodes running C++, C# .Net and Java are supported.
- GemFire supports peer-to-peer, client/server, and multi-site (WAN) architectures.

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com